



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

JAN

| APPLICATION NO. | FILING DATE | FIRST NAMED INVENTOR | ATTORNEY DOCKET NO. | CONFIRMATION NO. |
|-------------------------------|-------------|----------------------|---------------------|------------------|
| 10/025,774 | 12/26/2001 | Mauricio Lopez | BS01-321 | 5878 |
| 45695 | 7590 | 07/06/2005 | EXAMINER | |
| WITHERS & KEYS FOR BELL SOUTH | | | CHOW, CHIH CHING | |
| P. O. BOX 71355 | | | ART UNIT | PAPER NUMBER |
| MARIETTA, GA 30007-1355 | | | 2192 | |

DATE MAILED: 07/06/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

| | | |
|------------------------------|-----------------|--------------|
| Office Action Summary | Application No. | Applicant(s) |
| | 10/025,774 | LOPEZ ET AL. |
| | Examiner | Art Unit |
| | Chih-Ching Chow | 2192 |

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --
Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) Responsive to communication(s) filed on 01 May 2005.
- 2a) This action is **FINAL**. 2b) This action is non-final.
- 3) Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) Claim(s) 1-8 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) Claim(s) 9-12, 14-18, 20 and 21 is/are allowed.
- 6) Claim(s) _____ is/are rejected.
- 7) Claim(s) _____ is/are objected to.
- 8) Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) The specification is objected to by the Examiner.
- 10) The drawing(s) filed on 26 December 2001 is/are: a) accepted or b) objected to by the Examiner.
 Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
 Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
 a) All b) Some * c) None of:
 1. Certified copies of the priority documents have been received.
 2. Certified copies of the priority documents have been received in Application No. _____.
 3. Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- 1) Notice of References Cited (PTO-892)
- 2) Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
 Paper No(s)/Mail Date 05/29/02.
- 4) Interview Summary (PTO-413)
 Paper No(s)/Mail Date. 04/05/05.
- 5) Notice of Informal Patent Application (PTO-152)
- 6) Other: _____

DETAILED ACTION

1. This action is responsive to Request for Continued Examination dated May 1, 2005.
2. Per Applicants' request, claims 1, 5, 9, and 17 have been amended, claims 13 and 19 are canceled.
3. Claims 1-8 remain pending, claims 9-12, 14-18, 20-21 are allowable.

Response to Amendment

4. Applicants' amendment for Claims 1, 5, 9, and 13 have been fully considered respectfully by the examiner, claims 9 and 13 are allowable, claims 1, 5 remain pending.
5. The Examiner is maintaining the 35 USC § 102 and the 35 USC § 103 Rejections to claims 1-8. For the Applicants' convenience they are listed under items 8-13 of this Office Action, with the amendments requested by the Applicants.

Response to Arguments

6. Applicants' arguments for Claims 1-12, 14-18, 20-21 have been fully considered respectfully by the examiner.
7. Applicants' arguments are basically in the following points:
 - During the telephone interview on April 5, 2005, subject matter of the present application was discussed in relation to the current rejections based on the Dasgupta reference. In particular, the Examiner has recommended the Applicants to combine claim 5 into claim 1; to make claim 1 better conveying the teachings of the 'cross platforms' feature by using JAVA RMI and JAVA Virtual Machine. The Applicants have amended claims 9 and 17 as described above, therefore

claims 9-12, 14-18, and 20-21 are allowable (see Allowable Subject Matter below).

- The currently amended claim 1 recites:

“1. (Currently Amended) A method for modifying a computer application in substantially real-time without suspending or terminating the application, said method comprising the steps of:

- (a) using a system independent interface to connecting to an application executing on system independent platform on an application server, the application server having a computer memory;
- (b) through the system independent interface, acquiring program data that defines the underlying program structure including one or more object language components of the application;
- (c) displaying the program data including the one or more object language components to a maintenance person;
- (d) accepting a command from the maintenance person; and
- (e) through the system independent interface, executing the command to cause the program data of the executing application to be modified without suspending or terminating the executing application.”

Claim 5 recites:

“5. (Currently Amended) The method recited in claim 1, wherein the application is running upon a JAVA virtual machines the method further comprising the step of connection to the application using JAVA programming language RMI as the system independent interface.”

Examiner's Response: In response to applicant's argument that “claim 1 recites using a system independent interface to connect to an application executing on system independent platform on an application server, ... Thus, the claims are not directed to system dependent interfaces and platforms upon which the application executes, but

are instead directed to system independent ones.” (Remarks page 7, under ‘102 Rejections’). – ‘system independent interface’ is interpreted to mean an interface that is independent from a system. Based on this interpretation the added limitation “system independent interface” still reads on probe; probe is system independent because it can be used with VAX, UNIX, AIX, etc. . . . or other Operating Systems. As suggested by the Examiner during the phone interview on April 5, 2005, the independent claims need to recite both the ‘system independent interface’ and the feature of dependent claim 5 in order to distinguish over the teachings of Dasgupta. Since Applicant’s amendments to the claims have been entered and the Examiner has thoroughly addressed Applicant’s arguments without introducing any new grounds of rejection, a final rejection is considered appropriate. See MPEP 706.07 (b).

Claim Rejections - 35 USC § 102

8. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

9. Claims 1-4 are rejected under 35 U.S.C. 102(b) as being anticipated by “A Probe-Based Monitoring Scheme for an Object-Oriented, Distributed Operating System”, OOPSLA, 09/1986, by Partha Dasgupta, (hereinafter, “Dasgupta”).

CLAIM

1. A method for modifying a computer application in substantially real-time without suspending or terminating the application, said method comprising the steps of:

(a) using a system independent interface to connecting to an application executing on system independent platform on an application server, the application server

Dasgupta

Dasgupta teaches an example in using ‘PROBE’ for an application program. ‘Probing’ is a well-known computer software diagnostic technique to the people of ordinary skill in the art. Probe allows user to display and modify a designated program object while executing an application program (in real-time); the displaying and modifying is done at an

Art Unit: 2192

having a computer memory;

(b) through the system independent interface, acquiring program data that defines the underlying program structure including one or more object language components of the application;

(c) displaying the program data including the one or more object language components to a maintenance person;

(d) accepting a command from the maintenance person; and

(e) through the system independent interface, executing the command to cause the program data of the executing application to be modified without suspending or terminating the executing application.

independent window, which is an **user interface**, functioning as an **interactive console** for the user (*maintenance person*). Probing can display any designated variable contents, accepting arguments, it can also prompt user with a command line in order to modify a variable's content or any other commands. By modifying the variable's content, it can force the program to execute in an alternative path. See Dasgupta page 57, second paragraph, "The basic mechanism we propose, for monitoring, is the usage of probes. Probes are a powerful tool in many environments and has been proposed for deadlock detection, debugging, backup processing and so on" (*connecting to an application executing on an application server*), in Dasgupta page 57, first paragraph, "Give the ability to detect failing, flaky or failed components (software modules and hardware units) the system has the ability to reconfigure the healthy units, on the fly, to work around the faulty ones", also page 57, column 2, second paragraph, "The probe system also has some other payoffs like the easy implementation of interactive debugging support"; and on page 62, column 1 last paragraph, "The monitor periodically probes all the components in its list (*acquiring program data that defines the underlying program structure, displaying the program data*). The status of these components are stored in a fully replicated database. This database has the same structure and properties as the database used to locate Clouds objects,...it is highly available" -- probing works in real-time (without suspending or terminating the executing application), it interacts with the executing program and

2. The method of claim 1, further comprising the step of modifying data that has been cached in the computer memory.
3. The method recited in claim 1, further comprising the step of modifying an order of the execution of a plurality of methods within the application.
4. The method recited in claim 1, further

changes the program's behavior if the user has chose to do so (*accepting a command from the maintenance person, executing the command to cause the program behavior*); basically probing is able to give a snapshot of application program status (underlying program structure), it can manipulate the designated variable contents and thus change the program behavior, which is similar as described in the current application paragraph 0024, "these components are used to extract the fields and execute the methods of the object class. The fields of the object can be manipulated to create different behaviors in the object."

For the feature of claim 1 see claim 1 rejection. In Dasgupta, page 62, column 1, last paragraph, "The monitor periodically probes all the components in its list. The status of these components are stored in a **fully replicated database**. This database has the same structure and properties as the database used to locate Clouds objects,...it is **highly available**" to store the modifying data into cache memory for faster accessing is a design choice; it's an anticipated skill to those skilled in the art.

For the feature of claim 1 see claim 1 rejection. On Dasgupta page 57, under Section 2, second paragraph, "The basic building blocks in Clouds are **objects**, actions and process (*methods*). Processes are carriers of the thread of control, on behalf of actions. The actions are atomic units of activity, consisting of a partial **order of invocations of operations defined in objects**".

For the feature of claim 1 see claim 1

Art Unit: 2192

comprising the step of providing more detailed diagnostic messages in response to the command.

rejection. As mentioned in claim 1 rejection, PROBING allows user to set up commands at designated break points and to dump the program variable values, if the break points are set up appropriately, user can get more detailed diagnostic information when stepping through the break points.

Claim Rejections - 35 USC § 103

10. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

11. Claim 5 is rejected under 35 U.S.C. 103(a) as being unpatentable over "A Probe-Based Monitoring Scheme for an Object-Oriented, Distributed Operating System", by Partha Dasgupta (hereinafter, "Dasgupta"), in view of 'Introduction to Java Remote Method Invocation (RMI)', by Chris Matthews (hereinafter, "Matthews").

CLAIM

5. The method recited in claim 1, wherein the application is running upon a JAVA virtual machines the method further comprising the step of connection to the application using JAVA programming language RMI as the system independent interface.

Dasgupta / Matthews

For the feature of claim 1 see claim 1 rejection. Dasgupta teaches object-oriented, Distributed Operating System; Dasgupta mentioned in page 58, 4th paragraph 'object is an instance of an abstract data type' (*class*), 'set of routines that can access' (*method*) – these are all Java concepts, however, Java does not exist yet in 1986. Dasgupta teaches all aspect in claim 5 but does not mention the 'connection to the application using Java RMI' specifically. However Matthews teaches this feature in a prior art. In Matthews, page 1, 4th paragraph, "Java

RMI is shipped with the Java JDK 1.1 and higher. It is a true **distributed computing application interface for Java**".

It would have been obvious to a person of ordinary skill in the art at the time of the invention was made to supplement Dasgupta's disclosure of using probes by invoking a remote procedure taught by Matthews for the purpose of accessing remote objects over the network (see Matthews page 2, second paragraph).

12. Claim 6 is rejected under 35 U.S.C. 103(a) as being unpatentable over "A Probe-Based Monitoring Scheme for an Object-Oriented, Distributed Operating System", by Partha Dasgupta (hereinafter, "Dasgupta"), further in view of 'JAVA in A Nutshell' by David Flanagan (hereinafter "Flanagan").

CLAIM

6. The method recited in claim 1, further comprising the step of modifying an application written in an interpreter programming language.

Dasgupta / Flanagan

For the feature of claim 1 see claim 1 rejection. Dasgupta teaches all aspects of claim 1 but does not mention the 'interpreter programming language' specifically. However, Flanagan teaches that feature in his book. Java is an **interpreter programming language**, see O'reilly, page 14, last paragraph, "To invoke a Java program, you run the Java interpreter, java," and page 247, "The Java Interpreter" section.

It would have been obvious to a person of ordinary skill in the art at the time of the invention was made to supplement Dasgupta disclosure of the method using PROBing for Java language further taught by Flanagan for the purpose of invoking a Java program.

13. Claims 7-8 are rejected under 35 U.S.C. 103(a) as being unpatentable over "A Probe-Based Monitoring Scheme for an Object-Oriented, Distributed Operating System",

by Partha Dasgupta (hereinafter, "Dasgupta"), in view of 'Java Remote Method Invocation' 06/98, by Gopalan Suresh Raj (hereinafter "Raj"), and further in view of U.S. 6,681,389 by Norbert Engel et al. (hereinafter "Engel").

CLAIM

7. The method recited in claim 1, further comprising the steps of:
 - (a) accepting a selection of a method from the program data; and
 - (b) invoking the method from the command line with at least one new argument.

Dasgupta / Engel / Raj

For the feature of claim 1 see claim 1 rejection. Dasgupta teaches all aspects of claim 1 but does not mention the 'accepting a selection of method' and 'invoking method' (method is a Java term, Java does not exist yet in 1986) specifically. However, Engel teaches the 'accepting a selection' feature and Raj teaches 'invoking method' feature in an analogous art. In Engel, column 4, lines 61-65, "The user can select (there must be a list of software prompted for the user to do the selection) which application component to upgrade on one or more online machines/servers in the cluster. This provides the flexibility of updating a subset of application software rather than all application software on all machines/servers in a cluster." In Raj, page 1, 2nd paragraph, "In the RMI model, the server defines objects that the client can use remotely. The clients can now invoke methods of this remote object (*accepting a selection of a method*) as if it were a local object running in the same virtual machine as the client. RMI hides the underlying mechanism of transporting method arguments and return values across the network." Further, last paragraph, "which in turn calls the appropriate method on the server object. In other words, the stub acts as a proxy to the skeleton and the skeleton is a proxy to the actual remote method."

It would have been obvious to a person of ordinary skill in the art at the time of the

invention was made to supplement Dasgupta disclosure of the method using Probing by the feature of entering argument and invoking methods further taught by Engel and Raj for the purpose of passing parameters during method calls between machines (see Raj, page 2, 2nd paragraph).

8. The method recited in claim 1, further comprising the steps of:
 - (a) accepting a selection of a method and the program data;
 - (b) prompting the maintenance person for at least one new argument value; and
 - (c) invoking the method with the at least one argument.

For the feature of claim 1 see claim 1 rejection. For the rest of the features in claim 8, see claim 7 rejection.

Allowable Subject Matter

14. Claims 9-12, 14-18, 20 and 21 are allowable.
15. The following is a statement of reasons for the indication of allowable subject matter:

- The amended claims 9 and 17 specifically recited ‘running a JAVA virtual machine’ and ‘including a JAVA Remote Method Invocation (RMI)’, which implies that the application execution is running cross platforms, since both of JAVA virtual machine and JAVA RMI are well-known in the art with the feature of crossing platforms execution.
- The prior arts of record: Dasgupta, teaches using probe for a distributed operating system, while probe can monitor and modify application program execution without interrupting the current execution. However, Dasgupta’s disclosure does not teach using JAVA Virtual machine and JAVA RMI, which implies the cross

platforms execution. Matthews teaches JAVA RMI, mainly teaches using JAVA RMI to develop distributed computing applications. Flanagan teaches JAVA Virtual Machine. Engel and Raj teach the invoking JAVA methods. However, none of them, taken alone or in combination, teaches the following features in such a manner as recited in independent claims 9, and 17:

- A system for modifying an application insubstantially real-time during execution without suspending or terminating the application, wherein an application server running a JAVA virtual machine on which the application executes;
- an object shell console that attaches to the application through a JAVA RMI serving as a system independent interface while it is running to obtain program data defining the underlying program structure of the application including at least one object language component;
- a graphical user interface in the object shell console that is used to assist a maintenance person in modifying the program data of the application; and
- a command line for accepting a command to be executed, said command when executed will cause the execution of the application to be modified without suspending or terminating the application.

Conclusion

16. The following summarizes the status of the claims:

35 USC § 102 rejection: Claims 1-4

35 USC § 103 rejection: Claims 5-8

Allowable : Claims 9-12, 14-18, 20 and 21

Art Unit: 2192.

17. All claims are drawn to the same invention claimed in the application prior to the entry of the submission under 37 CFR 1.114 and could have been finally rejected on the grounds and art of record in the next Office action if they had been entered in the application prior to entry under 37 CFR 1.114. Accordingly, **THIS ACTION IS MADE FINAL** even though it is a first action after the filing of a request for continued examination and the submission under 37 CFR 1.114. See MPEP § 706.07(b). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire **THREE MONTHS** from the mailing date of this action. In the event a first reply is filed within **TWO MONTHS** of the mailing date of this final action and the advisory action is not mailed until after the end of the **THREE-MONTH** shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than **SIX MONTHS** from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Chih-Ching Chow whose telephone number is 571-272-3693. The examiner can normally be reached on 7:30am - 4:00pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam can be reached on 571-272-3695. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306. Any inquiry of a general nature of relating to the status of this application should be directed to the **TC2100 Group receptionist: 571-272-2100**.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status

Art Unit: 2192

information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Chih-Ching Chow

Examiner

Art Unit 2192

June 30, 2005

cc



**ANTONY NGUYEN-BA
PRIMARY EXAMINER**